<u>Amendments to the Claims</u>:

This listing of claims replaces all prior versions and listings of claims in the application:

<u>Listing of Claims</u>:

1.   (currently amended) An operating system, comprising:

a non-preemptive microkernel executing ~~one~~ <u>two</u> or more processes in accordance with a <u>non-preemptive scheduling scheme, wherein each process executed by the non-preemptive microkernel is only interrupted for a higher priority process to execute when the process blocks or explicitly requests to be preempted</u> ~~predetermined priority~~; and

one or more kernels each being executed as a process by the non-preemptive microkernel.

2.   (currently amended) The operating system of claim 1, wherein <u>at least</u> one of the <u>one or more</u> kernels executes an operating system as a dependent process.

3.   (currently amended) The operating system of claim 2, wherein the operating system is a time-sliced operating system or a <u>time-sliced</u> microkernel.

4.   (original) The operating system of claim 2, wherein the operating system is Unix.

5.   (currently amended) The operating system of claim 1, wherein each <u>of the two or more processes executed by the non-preemptive microkernel</u> ~~process~~ has its own stack.

6.   (currently amended) The operating system of claim 1, wherein <u>each of</u> the <u>two or more</u> processes <u>executed by the non-preemptive microkernel</u> communicate using one or more messages.

7.   (currently amended) The operating system of claim 1, wherein each <u>of the two or</u> <u>more processes executed by the non-preemptive microkernel</u> ~~process~~ has a unique process identifier (PID).

8.   (original) The operating system of claim 7, further comprising a mailbox coupled to a plurality of processes to service messages sent to a single PID.

9.   (currently amended) The operating system of claim 1, wherein <u>each of</u> the <u>two</u> <u>ore more</u> processes <u>executed by the non-preemptive microkernel</u> never <u>terminates</u> ~~terminate~~.

10.   (currently amended) The operating system of claim 1, wherein one of the <u>one or</u> <u>more</u> kernels is a microkernel.

11.   (currently amended) A method for operating a computer system <u>including a CPU</u>, comprising:
       managing ~~one~~ <u>two</u> or more processes with a non-preemptive microkernel, the microkernel ~~running~~ <u>executing</u> the ~~one~~ <u>two</u> or more processes in accordance with a <u>non-preemptive scheduling scheme, wherein each process executed by the non-preemptive</u> <u>microkernel is only interrupted for a higher priority process to execute when the process blocks</u> <u>or explicitly requests to be preempted</u> ~~predetermined priority~~; and
       executing one or more kernels as one or more processes managed by the non-preemptive microkernel.

12.   (currently amended) The method of claim 11, further comprising executing an operating system in one of the <u>one or more kernels</u> ~~microkernels~~ as a dependent process.

13.   (currently amended) The method of claim 12, wherein the operating system is a time-sliced operating system or a <u>time-sliced</u> microkernel.

14.   (original) The method of claim 12, wherein the operating system is Unix.

15.   (currently amended) The method of claim 11, wherein each ~~process~~ <u>of the two or more processes executed by the non-preemptive microkernel</u> has its own stack.

16.   (original) The method of claim 11, further comprising performing inter-process communication using one or more messages.

17.   (currently amended) The method of claim 11, wherein each ~~process~~ <u>of the two or more processes executed by the non-preemptive microkernel</u> has a unique process identifier (PID).

18.   (currently amended) The ~~operating system~~ <u>method</u> of claim 17, further comprising servicing messages sent to a single PID by a plurality of processes using a mailbox.

19.   (currently amended) The method of claim 11, further comprising executing the <u>two or more</u> processes without termination.

20.   (currently amended) The method of claim 11, further comprising executing a microkernel in one of the <u>one or more</u> kernels.

21.   (currently amended) A computer system, comprising:
       means for managing ~~one~~ <u>two</u> or more processes with a non-preemptive microkernel, the microkernel <u>executing</u> ~~running~~ the ~~one~~ <u>two</u> or more processes in accordance with a <u>non-preemptive scheduling scheme, wherein each process executed by the non-preemptive</u>

microkernel is only interrupted for a higher priority process to execute when the process blocks or explicitly requests to be preempted predetermined priority; and

    means for executing one or more kernels as one or more processes managed by the non-preemptive microkernel.

    22.   (currently amended) The system of claim 21, further comprising means for executing an operating system in one of the one or more kernels as a dependent process microkernels.

    23.   (currently amended) The system method of claim 22 12, wherein the operating system is a time-sliced operating system.

    24.   (currently amended) The system method of claim 22 12, wherein the operating system is Unix.

    25.   (currently amended) The system of claim 21, wherein each process of the two or more processes executed by the non-preemptive microkernel has its own stack.

    26.   (original) The system of claim 21, further comprising means for performing inter-process communication using one or more messages.

    27.   (currently amended) The system of claim 21, wherein each process of the two or more processes executed by the non-preemptive microkernel has a unique process identifier (PID).

    28.   (currently amended) The operating system of claim 21 [17], further comprising means for servicing messages sent to a single PID by a plurality of processes using a mailbox.

29.   (currently amended) The system of claim 21, further comprising means for executing each of the two or more processes executed by the non-preemptive microkernel without termination.

30.   (currently amended) The system of claim 21, further comprising means for executing a microkernel in one of the one or more kernels.

31.   (currently amended) A computer, comprising:

an interconnect bus;

one or more processors coupled to the interconnect bus and adapted to be configured for server-specific functionalities including network processing, file processing, storage processing and application processing;

a configuration processor coupled to the interconnect bus and to the processors, the configuration processor dynamically assigning processor functionalities upon request;

one or more data storage devices coupled to the processors and managed by a file system;

a non-preemptive microkernel executing two one or more processes in accordance with a non-preemptive scheduling scheme, wherein each process executed by the non-preemptive microkernel is only interrupted for a higher priority process to execute when the process blocks or explicitly requests to be preempted predetermined priority; and

one or more kernels each being executed as a process by the non-preemptive microkernel.

32.   (currently amended) The computer of claim 31, wherein the non-preemptive microkernel executes an operating system as a dependent process.

33.   (currently amended) The computer of claim 31, wherein the non-preemptive microkernel executes a network switch operating system as a dependent process.